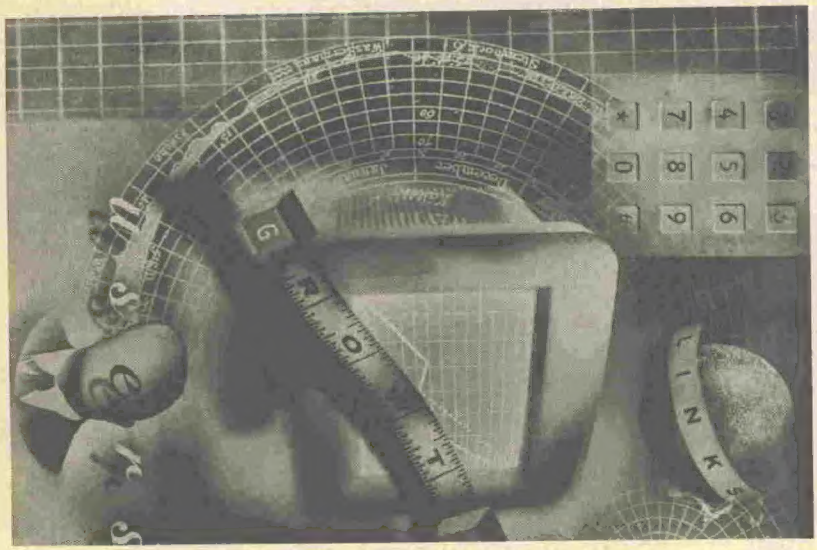
AutoCAD

ชัชวาล ศุภเกษม

เทคนิคการใช้ AutoCAD : คำสั่งช่วยเปลี่ยนค่าตัวอักษรแบบตัวเลข โดยเพิ่ม-ลด ค่าคงที่



นำเสนอคำสั่ง ADD ช่วยจัดการงานซ้ำ ๆ ในกรณีแก้ไขข้อความที่เป็นตัวเลข โดยเพิ่มหรือลดค่าด้วยขนาดค่าคงที่ค่าหนึ่ง ๆ

สำหรับผู้ใช้ AutoCAD กับงานเขียนแบบ การแก้ไขตัวอักษรที่เขียนจากคำสั่ง DTEXT หรือ TEXT ในการแก้ไขสามารถกระทำได้ครั้งละข้อความ โดยใช้คำสั่ง DDEDIT หรือ DDMODIFY ซึ่งจะให้ผู้ใช้เลือกข้อความที่ต้องการและคำสั่งจะแสดงไดอะล็อกบ็อกซ์ตามรูปที่ 1 และ 2 ตามลำดับ

ในบทความนี้จะนำเสนอคำสั่ง ADD ผ่านการโปรแกรม AutoLISP (สำหรับผู้อ่านที่ยังไม่เคยโปรแกรมด้วย AutoLISP ขอให้ศึกษาจากบทความในฉบับก่อน ๆ ในคอลัมน์ AutoCAD)

คำสั่งนี้จะช่วยจัดการงานซ้ำ ๆ ในกรณีแก้ไขข้อความที่เป็นตัวเลข โดยเพิ่มหรือลดค่าด้วยขนาดค่าคงที่ค่าหนึ่ง ๆ กระทำกับกลุ่มข้อความที่อยู่บน Layer เดียวกัน ตัวอย่างประยุกต์ใช้งาน เช่น ในแบบงานสถาปัตย์ที่มีการเปลี่ยนแปลงระดับของอาคาร เช่น ยกระดับทั้งอาคารอีก +0.50 เมตร หรือลดระดับทั้งหมด -0.5 เมตร ขอให้พิจารณาว่าในแบบอาคารหนึ่ง ๆ จะมีข้อความแสดงระดับพื้นมากเพียงใด (ดูรูปที่ 3 และ 4) หากต้องมานั่งแก้ไขค่าครั้งละตัว ๆ คงไม่ใช่วิธีการทำงานที่สุนทรีนัก หรือในงานเขียนแบบใด ๆ ที่มีการกำหนดลำดับเลขเรียงไปแล้ว แต่ต้องการเปลี่ยนตัวเลขเริ่มแรกของชุดเลขเรียงใหม่

สร้างไฟล์โปรแกรมและใช้งาน

สำหรับผู้อ่านที่ยังไม่เคยโปรแกรมด้วย AutoLISP ขอให้ศึกษาจากบทความในฉบับก่อน ๆ หัวข้อนี้จะขอกล่าวเพียงคร่าว ๆ เท่านั้น

การสร้างไฟล์โปรแกรม AutoLISP ใช้โปรแกรมใด ๆ ที่สามารถบันทึกไฟล์ในรูปรหัสแอสกีได้ ในการเตรียมบทความนี้ผู้เขียนเลือกใช้ NotePad บนวินโดวส์ ขอให้สร้างไฟล์ตามลิสติงที่ 1 บันทึกเป็นชื่อ ADD.LSP

โดยทั่วไปการโปรแกรม AutoLISP จะไม่มีผล Case sensitive แต่ในโค้ดนี้มีการใช้ฟังก์ชัน Initget และ Getkword ในบรรทัดที่ 62 และ 63 ตามลิสติงที่ 1 ขอให้พิมพ์ข้อความเป็นตัวอักษรพิมพ์ใหญ่-เล็กให้ตรงตามนี้ และในบรรทัดที่ 71 ให้ระวังเรื่องนี้ด้วย ก่อนใช้งานไฟล์โปรแกรม AutoLISP จะต้องมีการโหลดเข้าสู่หน่วยความจำก่อน โดยฟังก์ชัน LOAD บนบรรทัด Command: ใน AutoCAD เมื่อสร้างไฟล์และบันทึกเรียบร้อยแล้ว ให้กลับเข้าสู่โปรแกรม AutoCAD และพิมพ์ (LOAD "C:/LISP/ADD") ในกรณีที่บันทึกไว้ในพาธ C:\LISP ขอให้สังเกตว่าใช้เครื่องหมาย Slash (/) ในการกำหนดพาธ ต่างจากทั่วไปที่ใช้ Backslash (\)

ถ้าการโปรแกรมไม่มีข้อผิดพลาดจะพบกับข้อความ C:ADD บนบรรทัด Command: หากต่างจากนี้ให้กลับไปแก้ไขให้ถูกต้องและทำการโหลดใหม่ หรือศึกษาจากหนังสือคู่มือ AutoCAD ถึง Error message ที่เกิดขึ้นจากการโปรแกรม และแก้ไขในส่วนนั้น ๆ

พรอมต์ของคำสั่ง

เมื่อรันคำสั่ง ADD ที่บรรทัด Command: จะพบกับพรอมต์ดังต่อไปนี้

Command: พิมพ์ **add** แล้วกดแป้น Enter จะพบกับพรอมต์

Automatic addition or substraction text-number.

Addition increment <0.0>

พรอมต์นี้ให้ผู้ใช้ป้อนค่าเพิ่มหรือลดที่ต้องการ ขอให้สังเกตว่าใช้คำว่า Addition หมายความว่า ถ้าผู้ใช้ต้องการเพิ่มค่าตัวเลข ให้พิมพ์ค่าตัวเลขบวกโดยตรง คำสั่งจะนำค่าไปรวมกับตัวเลขค่าเดิมอยู่แล้ว ในกรณีต้องการจะให้เป็นการลดค่า ให้ป้อนค่าด้วยเลขลบแทน นอกจากนั้นสามารถรับค่าดีฟอลต์ในวงเล็บได้ โดยกด Enter ผ่าน

Decimal places <2>

ให้ผู้ใช้กำหนดจำนวนตำแหน่งทศนิยมที่ต้องการ รับเฉพาะจำนวนเต็มบวกและศูนย์ หากผู้ใช้ป้อนค่าลบ จะพบกับข้อความ Value must be positive. Try again: ให้ป้อนค่าใหม่ให้ถูกต้อง

Show plus (+) sign Yes/No: <Yes>

ใช้กำหนดให้แสดงเครื่องหมายบวกหรือลบ (+/-) นำหน้าตัวเลขได้ โดยป้อนค่า Y หรือ N สำหรับกรณีที่ต้องการและไม่ต้องการให้แสดงตามลำดับ

Select object on desired layer:

ให้เลือกหนึ่งวัตถุ ซึ่งเป็นวัตถุที่อยู่บน Layer เดียวกับกลุ่มข้อความที่ต้องการแก้ไขตัวเลขนี้ ไม่จำเป็นที่วัตถุนี้จะต้องเป็นข้อความแบบตัวเลขด้วย โปรแกรมจะทำการดักตรวจสอบ (Trapping) เอง

หากการเลือกไม่ถูกวัตถุใด ๆ เลย โปรแกรมจะยังคงแสดงพรอมต์นี้ให้ผู้ใช้เลือก เป็นเช่นนี้จนกว่าจะพบวัตถุ

หลังจากนั้นโปรแกรมจะทำการเปลี่ยนค่าให้ตามพารามิเตอร์ที่กำหนดข้างต้น และจะแสดงข้อความเป็น **Change text-number object = NNN** โดย NNN แทนจำนวนวัตถุที่เป็นข้อความตัวเลขที่ได้รับการเปลี่ยนค่าแล้วจากการทำงานของโปรแกรม และเมื่อทำงานจบจะแสดงข้อความ **Automatic change text-number objects on 0 layer, completely.** และจบการทำงานอย่างสมบูรณ์

แต่ในกรณีที่ไม่พบวัตถุที่เป็นข้อความตัวเลขบน Layer ที่เลือกเลย จะแสดงข้อความว่า **No text-number object found.** และจบการทำงาน

ข้อสังเกตการใช้คำสั่ง ADD

1. ใช้คำสั่ง ADD นี้ในการปัดเศษให้แสดงเป็นจำนวนทศนิยมที่มีตำแหน่งทศนิยมตามต้องการได้ โดยกำหนดค่าในพรอมต์ **Addition increment <0.0>** ให้มีค่าเป็นศูนย์ และที่พรอมต์ **Decimal places <2>** ให้กำหนดตำแหน่งทศนิยมที่ต้องการ

2. ในการโปรแกรมนี้ ไม่ได้ทำ Error trapping จากการกด ESC เพื่อยกเลิกคำสั่งไว้ เมื่อมีการกด ESC นี้จะเกิดการ Scrolling ของโค้ด ผู้อ่านสามารถปรับปรุงโปรแกรมโดยนิยามฟังก์ชัน *error* ใหม่ ขอให้ศึกษาได้จากบทความฉบับก่อน ๆ

3. ผู้เขียนได้ทดสอบโปรแกรมนี้บน AutoCAD รีลีส 14 อย่างไรก็ตาม เนื่องจากการโปรแกรมไม่ได้ใช้ฟังก์ชันของ AutoLISP ที่เพิ่มเติมมาใหม่เลย หรือกล่าวง่าย ๆ ว่าสามารถใช้กับรีลีสก่อน ๆ ได้ ทั้งรีลีส 12 และ 13

4. รูปวาดที่คำสั่งนี้จัดการแก้ไขค่าได้ คือวัตถุแบบ TEXT ซึ่งเขียนมาจากคำสั่ง TEXT หรือ DTEXT เท่านั้น คำสั่งนี้ไม่สามารถจัดการกับวัตถุแบบ ATTRIBUTE ได้ แม้ว่าจะแสดงเป็นลักษณะข้อความตัวเลขเช่นเดียวกัน

5. ใช้คำสั่ง C:ADD นี้ช่วยเติมเครื่องหมายบวก (+) นำหน้าค่าที่เป็นเลขบวกได้ โดยกำหนดค่าที่ Addition increment เป็น 0

Pseudocode

Pseudocode หรือแนวคิดลำดับการทำงานของโปรแกรม ADD มีดังนี้

1. รับค่าพารามิเตอร์ที่จะใช้กำหนดเป็นเงื่อนไขในการเพิ่มหรือลดค่า ได้แก่ ค่าคงที่ที่ใช้ในการเพิ่ม-ลด (#add), จำนวนหลักทศนิยม (#round), บังคับแสดงเครื่องหมายบวก-ลบ (#plus)

2. เลือกมาหนึ่งวัตถุ เพื่อให้โปรแกรมอ่านชื่อ Layer ที่ต้องการ

3. นำชื่อ Layer มาเลือกวัตถุบน Layer นี้ทั้งหมดเป็น Selection set (S)

4. อ่านวัตถุจาก Selection set มาทีละวัตถุ

4.1 ตรวจสอบว่าใช่ข้อความแบบตัวเลขหรือไม่

4.2 ถ้าใช่ ให้แปลงเป็นค่าตัวเลข แล้วปรับเพิ่ม-ลดค่าตามพารามิเตอร์ข้อ 1 แล้วนำค่าใหม่ไปอัพเดตฐานข้อมูลของรูปวาดนี้ แต่ถ้าไม่ใช่ให้ไปอ่านวัตถุใหม่เข้ามา และทำเช่นนี้จนครบทุกวัตถุใน Selection set (S)

อธิบายโปรแกรม

บรรทัดที่ 1-3 เป็นบรรทัด comments ทำได้โดยขึ้นต้นบรรทัดด้วยเครื่องหมาย ; (semicolon) ใช้ในการบรรยายหรือบันทึกช่วยเตือนความจำในการโปรแกรมหนึ่ง ๆ

บรรทัดที่ 5-6 นิยามสับรูทีน dxf ใช้อ่านค่าของ Sublist ในฐานข้อมูลของ AutoCAD จะมีการบันทึกลักษณะประจำตัวของรูปวาดหนึ่ง ๆ ไว้ ขอยกตัวอย่างสำหรับรูปวาดแบบข้อความจะเก็บข้อมูลดังนี้

```
(-1 . <Entity name: 25d0500>)
; หมายเลขประจำตัวของรูปวาด เรียกว่า Entity name
(0 . "TEXT")
; ประเภทของรูปวาด ในที่นี้คือ TEXT (ข้อความ)
(5 . "20")
; Entity handle เลขประจำตัวอย่างหนึ่ง
(100 . "AcDbEntity")
; Subclass data marker
(67 . 0)
; อยู่ในภาวะ Paper Space (1) หรือ Model Space (0)
(8 . "0")
; ชื่อ Layer
(100 . "AcDbText")
; Subclass data marker
(10 82.4194 133.532 0.0)
; ตำแหน่งอ้างอิงที่เขียนข้อความ
(40 . 10.0)
; ความสูงตัวอักษร
(1 . "TEST FOR TEXT.")
; ข้อความตัวอักษรที่เขียน
(50 . 0.0)
; มุมหมุน
(41 . 1.0)
; ตัวคูณ X-scale สำหรับแบบ Fit
(51 . 0.0)
; ค่ามุมเอียงของตัวอักษร
(7 . "STANDARD")
; ชื่อ Style ที่ใช้
(71 . 0)
; ลักษณะการสร้างข้อความ ค่า 2 และ 4 สำหรับการกลับด้านแบบสะท้อนกระจก และแบบกลับบนลงล่างตามลำดับ
(72 . 0)
; ทิศทางการวางอักษร แบบ Left, Center, Right
(11 0.0 0.0 0.0)
; จุดของ alignment ถ้าค่าของ 72 และ 73 ไม่เป็นศูนย์
(210 0.0 0.0 1.0)
; Extrusion vector กำหนดระนาบที่วัตถุอยู่
(100 . "AcDbText")
; Subclass data marker
(73 . 0) )
; ทิศทางการวางตัวอักษรในแนวดิ่ง
```

บรรทัดที่ 8-14 กำหนดค่าเริ่มต้นให้กับค่าดีฟอลต์ของตัวแปรชื่อ #add, #round และ #plus

บรรทัดที่ 16-22 นิยามสับรูทีน no_chk ใช้ตรวจสอบว่ารูปวาดเป็นข้อความแบบตัวเลขหรือไม่ ถ้าเป็นให้ค่าตัวเลขนั้นออกมา มิฉะนั้นให้ผลลัพธ์เป็น nil ดูรูปที่ 5 ประกอบ

บรรทัดที่ 24-35 นิยามสับรูทีน decimal ใช้หาจำนวนตำแหน่งทศนิยมของข้อความแบบตัวเลข ให้ผลลัพธ์เป็นจำนวนเต็มแทนจำนวนตำแหน่งทศนิยม ดูรูปที่ 6 ประกอบ

บรรทัดที่ 37-51 นิยามสับรูทีน digit_compensate ใช้ชดเชยจำนวนตำแหน่งทศนิยม ในกรณีที่น้อยกว่าค่ากำหนด #round ให้เติมท้ายด้วยเลขศูนย์จนครบ แต่ถ้าจำนวนทศนิยมมากกว่าอยู่แล้ว ไม่ต้องทำอะไรให้จำนวนนั้นออกมาเลย ดูรูปที่ 7 ประกอบ

บรรทัดที่ 53-99 นิยามเมนโปรแกรม C:ADD เริ่มจากบรรทัดที่ 54 ใช้ฟังก์ชัน defun นิยามคำสั่ง C:ADD โดยกำหนดตัวแปร local เป็น tmp, e, i, j, s, no, ei และ new และจบการนิยามด้วยวงเล็บปิดในบรรทัดที่ 99

บรรทัดที่ 55 แสดงข้อความ Automatic addition or substraction text-number. ที่บรรทัด Command บนหน้าจอใน AutoCAD เพื่อให้ผู้ใช้รู้ว่าคำสั่งนี้ใช้ทำอะไร

บรรทัดที่ 56-57 แสดงพรอมต์ Addition increment < ค่าดีฟอลต์ #add> โดยใช้ฟังก์ชัน getdist เพื่อให้ผู้ใช้ป้อนค่าจำนวนจริงที่จะเพิ่ม ในกรณีที่ผู้ใช้กด Enter ผ่านเฉย ๆ ตัวแปร tmp จะมีค่าเป็น nil ทำให้ใช้ค่าดีฟอลต์ แต่ถ้ามีค่าต่างจาก nil ให้เก็บ (hold) ค่าใหม่จาก tmp ไว้ในตัวแปร #add

บรรทัดที่ 58-60 เช่นเดียวกับบรรทัดที่ 56-57 แต่แสดงพรอมต์เป็น Decimal places < ค่าดีฟอลต์ #round> และใช้ฟังก์ชัน getint เพื่อให้ผู้ใช้ป้อนค่าจำนวนเต็มกำหนดจำนวนทศนิยม และใช้ฟังก์ชัน initget กับอาร์กิวเมนต์ค่า 4 เพื่อป้องกันการป้อนค่าเป็นลบ

บรรทัดที่ 58-60 เช่นเดียวกับบรรทัดที่ 56-57 แต่แสดงพรอมต์เป็น Show plus (+) sign < ค่าดีฟอลต์ #plus> และใช้ฟังก์ชัน getkword เพื่อให้ผู้ใช้ป้อน อักษร Y, Ye, Yes สำหรับการตอบ Yes หรือ N, No สำหรับการตอบ No โดยบังคับข้อความป้อนด้วยฟังก์ชัน initget กับอาร์กิวเมนต์ Yes และ No

บรรทัดที่ 64 ใช้ฟังก์ชัน While บังคับ loop แสดงพรอมต์ Select object on desired layer: ควบคุมให้ผู้ใช้เลือกจนกว่าจะได้หนึ่งวัตถุแล้วค่อยหลุดจาก loop นี้ และเก็บไว้ที่ตัวแปร e

บรรทัดที่ 65 กำหนดค่าเริ่มต้นให้กับดัชนี i และ j เป็น -1 และ 0 ตามลำดับ เพื่อใช้ใน Repeat loop ต่อไป

บรรทัดที่ 66 ที่บรรทัด Command: บนหน้าจอให้ขึ้นบรรทัดใหม่เพื่อพร้อมแสดงข้อความใหม่จากบรรทัดที่ 84 และ 85

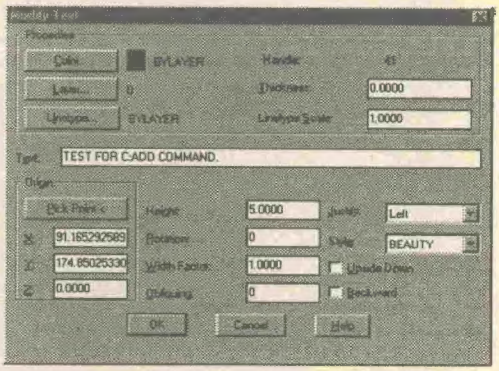
บรรทัดที่ 67-88 ขอให้สังเกตการทำงานจากผังการทำงานใน รูปที่ 8

บรรทัดที่ 89-97 ใช้ฟังก์ชัน if ตรวจสอบเงื่อนไขว่าได้พบข้อความตัวเลขบน Layer ที่เลือกหรือไม่ โดยตรวจจากค่าดัชนี j ถ้าไม่มีหรือค่าเป็นศูนย์ ให้แสดงข้อความว่า No text-number object found. แต่ถ้าพบ หรือคือค่า j ไม่เป็นศูนย์ ให้แสดงข้อความว่า Automatic change text-number objects on **ชื่อ-Layer** layer, completely. (หมายถึงว่าได้เปลี่ยนค่าของข้อความตัวเลขที่พบบนชื่อ Layer นั้น ๆ เสร็จสมบูรณ์แล้ว)

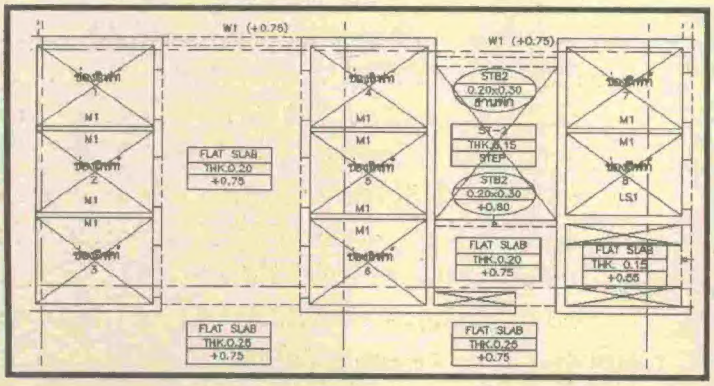
บรรทัดที่ 98 ฟังก์ชัน princ เมื่อใช้แบบไม่มีอาร์กิวเมนต์ ปกติใช้สำหรับการจำกัดไม่ให้แสดงผลลัพธ์สุดท้ายจากการทำงานของฟังก์ชันใด ๆ ที่บรรทัด Command:

บรรทัดที่ 99 วงเล็บปิดของฟังก์ชัน defun ในบรรทัดที่ 54

รูปที่ 1 ไดอะล็อกบ็อกซ์คำสั่ง DDEDIT ใช้แก้ไขวัตถุแบบตัวอักษร จากคำสั่ง DTEXT หรือ TEXT



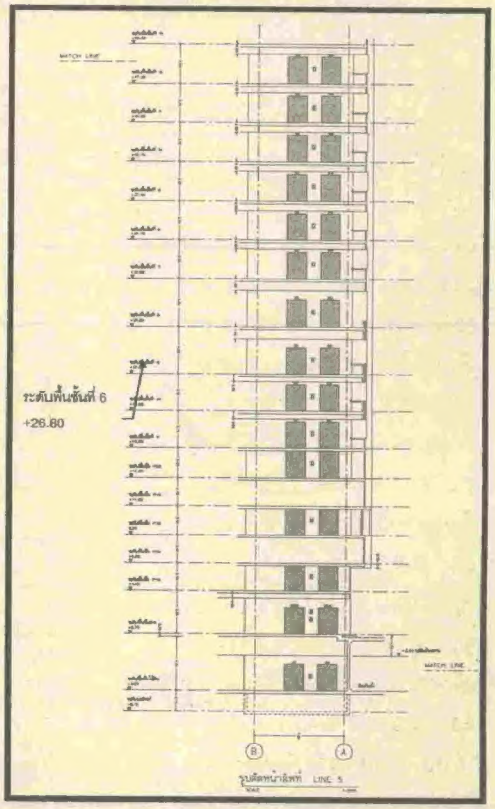
รูปที่ 2 ไดอะล็อกบ็อกซ์คำสั่ง DDMODIFY ใช้แก้ไขวัตถุแบบต่าง ๆ



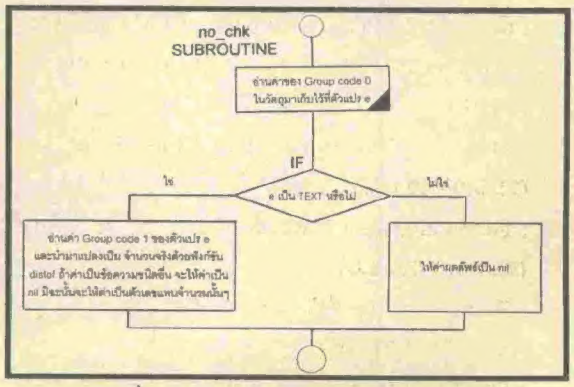
รูปที่ 3 แบบแปลนงานสถาปัตย์ แสดงค่าระดับพื้น เมื่อมีการเปลี่ยนแปลงระดับ ใช้คำสั่ง ADD ช่วยแก้ไข

ก้าวต่อไปของการใช้งาน

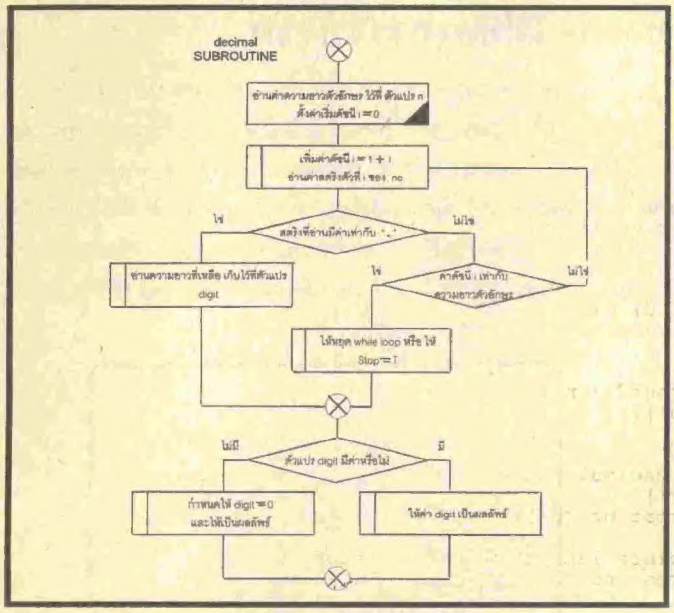
เพื่อความสะดวกของการใช้คำสั่ง ADD เราสามารถกำหนดให้โปรแกรมถูกโหลดเองทุกครั้งของการเปิดไฟล์ได้ โดยการวางบรรทัด (LOAD "C:/LISP/ADD") ไว้ในไฟล์ ACAD.LSP นอกจากนั้นยังสามารถกำหนดเป็นบัททอนเมนูสำหรับบริลสบนวินโดวส์ เพื่อให้เรียกใช้คำสั่งได้ง่าย ๆ โดยการปรับแต่งจากคำสั่ง TOOLBAR หรือกำหนดให้เรียกจากเมนูได้ ขอให้ผู้อ่านศึกษาเพิ่มเติมได้จากหนังสือคู่มือของ AutoCAD เอง ในส่วน Customization Guide



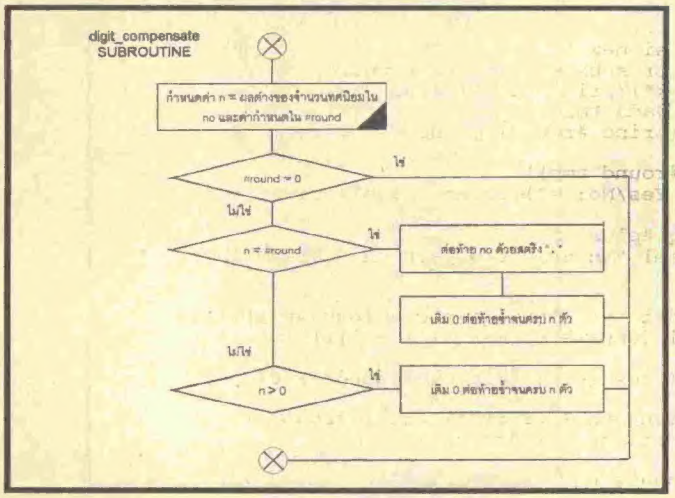
รูปที่ 4 แบบแสดงภาคตัด (Section) ของแบบอาคารหนึ่งที่กำหนดระดับความสูง และมีการเปลี่ยนค่า



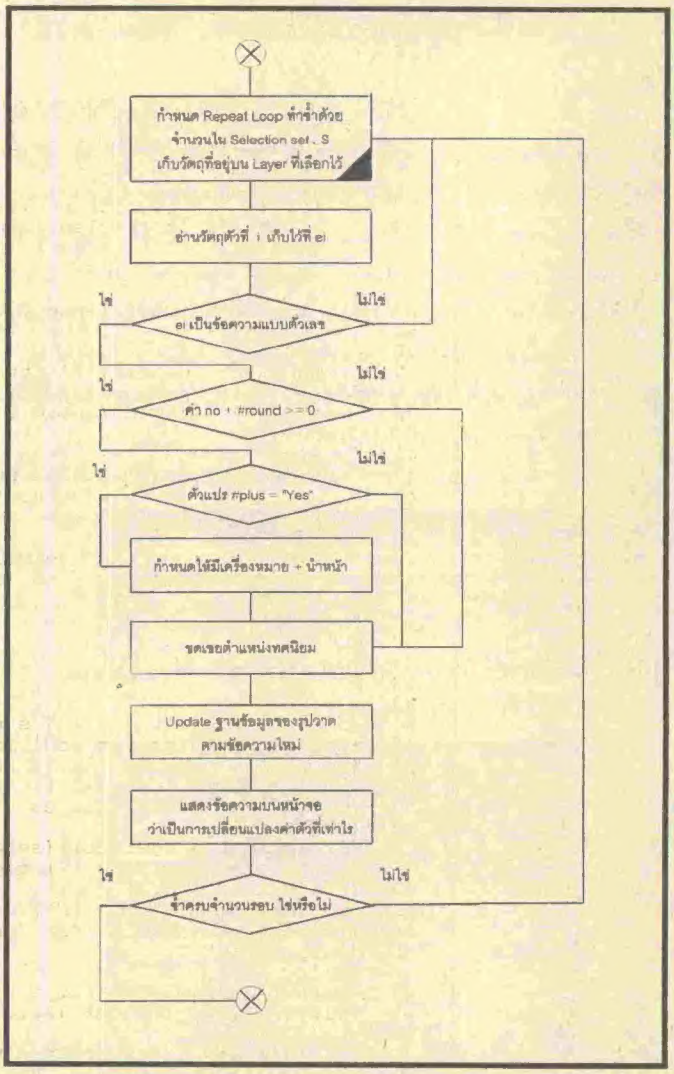
รูปที่ 5 ผังการทำงานของสับรูทีน no_chk



รูปที่ 6 ผังการทำงานของสับรูทีน decimal



รูปที่ 7 ผังการทำงานของสับรูทีน digit_compensate



รูปที่ 8 ผังการทำงานของเมนโปรแกรม C:ADD ในบรรทัดที่ 67-88

ลิสติงที่ 1
ไฟล์โปรแกรม
ADD.LSP

```
1  ;;; Automatic change by addition/substraction text-number objects
2  ;;; on the selected layer.
3  ;;; Developed by S.Chatchawal, 1997
4
5  ;; Read dxf
6  (defun dxf ( code e ) (cdr (assoc code (entget e))))
7
8  ;; Initialize default value
9  ;; Addition value
10 (if (not #add)(setq #add 0.0))
11 ;; Round off decimal place
12 (if (not #round)(setq #round 2))
13 ;; Plus sign
14 (if (not #plus)(setq #plus "Yes"))
15
16 ;; Subroutine for check an object is text-number or not?
17 (defun no_chk ( ei )
18  (if (= "TEXT" (dxf 0 ei))
19    (distof (dxf 1 ei) 2)
```

→

ลิสติงที่ 1 (ต่อ)

```
    'nil
  )
)

;; Count digits of decimal number
;; no is text-number
(defun decimal ( no / stop i digit )
  (setq n (strlen no) i 0)
  (while (not stop)
    (if (= "." (substr no (setq i (1+ i)) 1))
     (setq digit (strlen (substr no (1+ i) (- n i))))
     (if (= i n)(setq stop 'T))
    )
  )
  (if (= 'nil digit) (setq digit 0) digit)
)

;; Compensate decimal digits
(defun digit_compensate ( no #round / n )
  (setq n (- #round (decimal no)))
  (cond
    ((= #round 0) 'T)
    ((= n #round) ; no has not decimal place
        (setq no (strcat no "."))
        (repeat n (setq no (strcat no "0")))
    )
    ((> n 0) ; no has decimal place less than #round setup
        (repeat n (setq no (strcat no "0")))
    )
  )
 no
)

;; Main
(defun c:add (/ tmp e i j s no ei new )
 (princ "\n Automatic addition or substraction text-number.")
 (princ "\n Addition increment <")(princ #add)(princ "> ")
 (if (setq tmp (getdist))(setq #add tmp))
 (princ "\n Decimal places <")(princ #round)(princ "> ")
 (initget 4)
 (if (setq tmp (getint))(setq #round tmp))
 (princ "\n Show plus (+) sign Yes/No: <")(princ #plus)(princ "> ")
 (initget "Yes No")
 (if (setq tmp (getkword))(setq #plus tmp))
 (while (not (setq e (car (entsel "\n Select object on desired layer: ")))))
 (setq i -1 j 0)
 (princ "\n")
 (repeat (sslength (setq s (ssget "x" (list (assoc 8 (entget e))))))
  (if (setq no (no_chk (setq ei (ssname s (setq i (1+ i))))))
    (progn
      (if (>= (atof (setq new (rtos (+ no #add) 2 #round))) 0)
        (if (= "Yes" #plus)
          (setq new (digit_compensate (strcat "+" new) #round))
          (setq new (digit_compensate new #round))
        )
        (setq new (digit_compensate new #round))
      )
      (entmod
        (subst
         (cons 1 new)
         (assoc 1 (entget ei))
         (entget ei)
        )
      )
      (princ "\r Change text-number object = ")
      (princ (setq j (1+ j)))
    )
  )
 )
 (if (= j 0)
   (princ "\n No text-number object found.")
   (princ
    (strcat "\n Automatic change text-number objects on "
        (dxf 8 e)
        " layer, completely."
    )
   )
 )
 (princ)
)
```